DOCKET NO.: MSFT-0174/150793.01
Application No.: 09/557,250
Office Action Dated: February 24, 2006

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

REMARKS

According to the present Office Action, dated February 24, 2006, claims 1-13, 16-20, and 22-27 are pending in the Application, and these same claims stand rejected. At the outset, the Applicants want to thank the Examiner for considering the remarks made in the previous response to the non-final Office Action. In the present response, Applicants address the obviousness and double patenting rejections.

Claim Rejections – 35 U.S.C. § 103(a)

Claims 1, 16, and 22 are the independent claims. Claim 1, for example, recites the following subject matter:

> A method for utilizing applications' state dependency information to efficiently perform a backup service operation in a computer system, comprising the acts of:
>> registering applications loaded in said computer system with *an application dependency application programming interface (API) for communications of application's state dependency information among applications*, a common software agent, a storage component utilized by said agent and a backup service;
>> storing in said storage component at least one application's state dependency information; and
>> communicating said at least one application's state dependency information from said storage component to said backup service.

(emphasis added). Specifically, claim 1 recites "an application dependency application programming interface *(API) for communications of application's state dependency information among applications*."

In the Office Action, in order to address this limitation, the Examiner cites col. 6, ll. 41-58 and col. 6, ll. 32-45 of the Lomet reference (U.S. Patent No. 5,870,763), along with col. 114, l. 5 – col. 115, l. 17) of the Van Huben et al. reference (U.S. Patent No. 5,920,873). *See* Office Action, Feb. 24, 2006, pp. 11-12. The Lomet reference merely discloses an aspect of the disclosure that optimizes the application read operation to avoid writing the object data read to the log record. The read optimizing technique eliminates posting the read values to the log by substituting, for the read values, an identity of the location from where the values are read and posting the identity instead of the values. Moreover, Lomet discloses a cache manager that has an object table which tracks the objects maintained in a volatile cache. The object table includes field to track dependencies among the objects.

DOCKET NO.: MSFT-0174/150793.01
Application No.: 09/557,250
Office Action Dated: February 24, 2006

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

However, "an (API) for communications of application's state dependency information among applications" (claim 1) is something wholly distinct from fields that track dependencies among object. There is simply no motivation to combine the APIs from the Van Huben et al. reference with the object tables of Lomet. Such a combination does not make any technological sense, for it entails that API's would have to reside in object tables (a la the fields). A fortiori, such a combination would destroy the intended functionality of the claimed subject matter, for it would embed API's inside objects—such as tables.

To the Examiner's credit, the following combining motivation is provided in the Office Action:

> It would have been obvious to one of ordinary skill in the art at the time the inventions was made to combine the teachings of Lomet with the teachings of Van-IBM *in order to facilitate using application programming interface (API) because the API would support providing software modules.*

(Office Action, p. 12) (emphasis added). While it may be true, in general terms, that APIs can "support providing software modules," the Applicants disagree that "[APIs] for communications of application's state dependency information among applications" can somehow be embedded in object tables – and not only that, but function in such a way that "for each object entry, a predecessor field … lists all objects that must be flushed ...." (col. 6, ll. 53-56). In short, the Applicants submit that a field is not a substitute for an API in the context discussed above.

Independent claims 16 and 22 recite similar subject matter: "said API enables an agent to collect, store and package information about state dependencies among applications in response to a request by a service" (claim 16); "an agent that functions according to communication protocols of an application programming interface (API) in said system for processing said dependency information, wherein said dependency information includes information about dependencies among applications executing on the system" (claim 22).

Dependent claims 2-13, 17-20, and 23-27, depend either directly or indirectly from independent claims 1, 16, and 22, respectively, and are therefore deemed patentable for similar reasons.

DOCKET NO.: MSFT-0174/150793.01
Application No.: 09/557,250
Office Action Dated: February 24, 2006

PATENT
REPLY FILED UNDER EXPEDITED
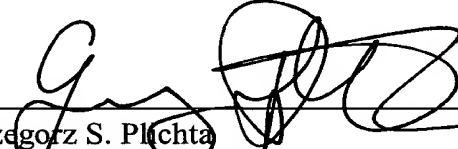PROCEDURE PURSUANT TO
37 CFR § 1.116

Double Patenting

In light of remarks made above, Applicants submit that they have obviated the double patenting rejections. Specifically, because the limitation of "an (API) for communications of application's state dependency information among applications" cannot be found in the art cited in the present Office Action, the current double patenting rejection should be withdrawn.

CONCLUSION

Applicants believe that the present Amendment is responsive to each of the points raised by the Examiner in the Office Action, and submits that Claims 1-13, 16-20, and 22-27 pending in the application are in condition for allowance. Favorable consideration and passage to issue of the application at the Examiner's earliest convenience is earnestly solicited.

Date: April 24, 2006

Grzegorz S. Plichta
Registration No. 55,541

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439